Remarks/Arguments

The Applicants respectfully request further examination and reconsideration in view of the amendments made above and the comments set forth below. Claims 1-45, 47-52, and 59-71 were pending. Claims 46 and 53-58 were previously canceled. Within the Office Action, Claims 1-45, 47-52, and 59-71 have been rejected under 35 U.S.C. § 103(a). By way of the above amendments, Claims 1-4, 14-18, 25-27, 36, 48, 49, 59, 63, 70, and 71 have been amended. Accordingly, Claims 1-45, 47-52 and 59-71 are now pending.

Rejections under 35 U.S.C. § 103(a)

Claims 1-5, 9, 11, 12, 19, 20, 26-28, 31, 36, 37, 39, 42, 43, 48-50, and 61-69

Within the Office Action, Claims 1-5, 9, 11, 12, 19, 20, 26-28, 31, 36, 37, 39, 42, 43, 48-50, and 61-69 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 7,313,694 to Riedel ("Riedel") in view of Zadok, "Cryptfs: A Stackable Vnode Level Encryption File System" ("Zadok"). The Applicants respectfully disagree.

Riedel is directed to a technique for secure file access control via directory encryption. Riedel discloses encrypting filenames to protect them in the event a server is untrustworthy, such as in a distributed computing environment. Riedel also discloses encrypting filenames in a directory structure without otherwise changing the directory structure. (Riedel, Abstract)

Zadok discloses a "stackable" vnode interface. As Zadok explains in its section 1.1, "With vnode stacking, several vnode interfaces may exist and may call each other in sequence: the code for a certain operation at stack level N typically calls the corresponding operation at level N-1, and so on." Zadok further explains: "Cryptfs is designed to be simple in principle. The file system interposes (mounts) itself on top of any directory, encrypts file data before it is passed to the interposed-upon file system, and decrypts in the reverse direction." (Zadok, page 2, col. 2, first full paragraph)

As shown in Figure 1 of Zadok, the Cryptfs layer is separate from the vnode layer; it is a separate module whose encryption components are not integrated with the vnode layer. Cryptfs is a modular file system that is mounted in the kernel but is not integrated with the vnode. Cryptfs is a feature inside a kernel that works with low level functions of the kernel to provide the Cryptfs file system. As Zadok explains in its section 1.1, third paragraph, "system calls are translated into vnode level calls, and those invoke their Cryptfs equivalents."

The Office Action does not dispute that neither Zadok nor Riedel, either alone or in combination, discloses a kernel that comprises a virtual node that encrypts and decrypts data.

As discussed below, claims in this application recite a kernel that comprises a virtual node that encrypts and decrypts data. The Office Action does not dispute that the prior art does not teach this element.

As for Riedel, it is admitted in the Office Action that "Riedel does not expressly mention the kernel of the operating system, but it is inherent that the operating system has a kernel." While the system in Riedel may inherently include a kernel, it does not disclose one that comprises a vnode to decrypt and encrypt data as recited in the claims of this application.

As for Riedel modified by Zadok, it is stated at pages 4-5 of the Office Action:

> Riedel as modified further teaches:
>
> b. A virtual node configured (cryptfs *is interfaced with* the vnode in the vnode layer to implement the file security system that includes encryption and decryption . . . to decrypt and encrypted directory entry to determine a location of the encrypted data file (decryption of the filename and the i-node pointer) . . . and to decrypt the encrypted data file to access data contained therein (data files can be encrypted for added security). [italics added and citations omitted]

The Office Action recognizes that an encryption and decryption element–cryptfs–is *interfaced* with the vnode. *It is not a part of the vnode itself.* Thus, Riedel, modified by Zadok, does not disclose a *kernel that comprises a virtual node* to encrypt and decrypt data, as recited in the claims of the application.

Further, one skilled in the art would not modify Zadok to integrate encryption and decryption in the vnode. Doing so would violate Zadok's design goals, as stated in its section 2:

> Cryptfs is designed to be simple. . . . . Our explicit design goals were:
>
> . . .
>
> - Portability: Cryptfs should be more portable than other kernel based file systems, by using a stackable vnode interface. It should not require modification to other file systems or user applications, and it should keep the underlying file system valid.

Zadok achieves it goals using stackable vnodes, not integrated vnodes as in accordance with embodiments of the presently claimed invention.

Embodiments

In accordance with embodiments of the presently claimed invention, a virtual memory facility is modified so that all incoming data is decrypted and all outgoing data is encrypted: a vnode in accordance with the embodiments is modified to encrypt and decrypt data entering and leaving kernel space. In one embodiment, "encryption drivers are *integrated into* the vnode interface structure" of UNIX source code. (Present Specification at page 45, lines 17-18; see also page 59, lines 24-25; italics added)

As further differences, Cryptfs uses user sessions to protect against cloning user IDs (UIDs), whereas embodiments of the presently claimed invention use i-node information such as the credentials of a user, which may or may not include UIDs.

The independent Claim 1 is directed to a computer system comprising a memory portion containing an encrypted data file and an operating system comprising a kernel. The kernel of Claim 1 comprises a virtual node to decrypt an encrypted directory entry to determine a location of the encrypted data file and to decrypt the encrypted data file to access data contained therein. Neither Riedel nor Zadok, either alone or in combination, discloses a virtual node to decrypt an encrypted directory entry to determine a location of the encrypted data file and to decrypt the encrypted data file to access data contained therein, as recited in Claim 1. For at least these reasons, the independent Claim 1 is allowable over Riedel, Zadok, and their combination.

Claims 2-5, 9, 11, 12, 19, 20, 61, and 62 all depend on the independent Claim 1. As explained above, the independent Claim 1 is allowable over Riedel, Zadok, and their combination. Accordingly, Claims 2-5, 9, 11, 12, 19, 20, 61, and 62 are all also allowable as depending on an allowable base claim.

The independent Claim 26 is directed to a computer system comprising a first device and a second device. The first device has an operating system kernel and a directory structure with directory information comprising encrypted data file names and corresponding encrypted data file locations for accessing encrypted data files within a file system. The operating system kernel is to decrypt the encrypted data file names and encrypted data file locations using one or more encryption keys to recover clear data corresponding to the data file names, data file locations, and data files. The operating system kernel comprises a virtual node to encrypt the clear data using the one or more encryption keys to generate cipher data corresponding to the directory information and encrypted data files. The second device is coupled to the first device to exchange cipher data with the first device. Neither Riedel nor Zadok, either alone or in combination, discloses an operating system kernel that comprises a virtual node to encrypt the

clear data using the one or more encryption keys to generate cipher data corresponding to the directory information and encrypted data files, as recited in Claim 26. For at least these reasons, the independent Claim 26 is allowable over Riedel, Zadok, and their combination.

Claims 27, 28, 31, and 63-65 all depend on the independent Claim 26. As explained above, the independent Claim 26 is allowable over Riedel, Zadok, and their combination. Accordingly, Claims 27, 28, 31, and 63-65 are all also allowable as depending on an allowable base claim.

The independent Claim 36 is directed to a method of storing an encrypted data file in a computer file system having a directory. The method of Claim 36 comprises receiving a clear data file having a name and executing kernel code in an operating system, the kernel code comprising a virtual node integrated with drivers to encrypt the clear data file to generate an encrypted data file using a symmetric key, store the encrypted data file at a location in the computer file system, and store in the directory an entry containing an encryption of the name and an encryption of the location. Neither Riedel nor Zadok, either alone or in combination, discloses kernel code that comprises a virtual node integrated with drivers to encrypt the clear data file to generate an encrypted data file using a symmetric key, store the encrypted data file at a location in the computer file system, and store in the directory an entry containing an encryption of the name and an encryption of the location, as recited in Claim 36. For at least these reasons, the independent Claim 36 is allowable over Riedel, Zadok, and their combination.

Claims 37, 39, 42, 43, 66, and 67 all depend on the independent Claim 36. As explained above, the independent Claim 36 is allowable over Riedel, Zadok, and their combination. Accordingly, Claims 37, 39, 42, 43, 66, and 67 are all also allowable as depending on an allowable base claim.

The independent Claim 48 is directed to a computer system that comprises a processor, a physical memory containing an encrypted data file and a directory, a secondary device coupled to the physical memory, and an operating system comprising a kernel. The directory comprises a record having a first element corresponding to an encrypted name of the data file and a second element corresponding to an encrypted location of the data file in the memory. The kernel comprises a virtual node integrated with drivers to decrypt the first and second elements to access the encrypted data file from memory when transferring the data file from the memory to the secondary device and to re-encrypt the first and second elements when transferring the data file from the secondary device to the memory. Neither Riedel nor Zadok, either alone or in combination, discloses a kernel that comprises a virtual node integrated with drivers to decrypt

the first and second elements to access the encrypted data file from memory when transferring the data file from the memory to the secondary device and to re-encrypt the first and second elements when transferring the data file from the secondary device to the memory, as recited in Claim 48. For at least these reasons, the independent Claim 48 is allowable over Riedel, Zadok, and their combination.

Claims 49, 50, 68, and 69 all depend on the independent Claim 48. As explained above, the independent Claim 48 is allowable over Riedel, Zadok, and their combination. Accordingly, Claims 49, 50, 68, and 69 are all also allowable as depending on an allowable base claim.

Claims 6-8, 14, 15, 29, 38, 39, 51, and 52

Within the Office Action, Claims 6-8, 14, 15, 29, 38, 51, and 52 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Riedel in view of Zadok as applied to claim 1, and further in view of U.S. Patent Pub. No. 2003/0005300 to Noble et al. ("Noble"). The Applicants respectfully disagree.

Claims 6-8, 14, and 15 all depend on the independent Claim 1. As explained above, the independent Claim 1 is allowable. Accordingly, Claims 6-8, 14, and 15 are all also allowable as depending on an allowable base claim.

Claim 29 depends on the independent Claim 26. As explained above, the independent Claim 26 is allowable. Accordingly, Claim 29 is also allowable as depending on an allowable base claim.

Claim 38 depends on the independent Claim 36. As explained above, the independent Claim 36 is allowable. Accordingly, Claim 38 is also allowable as depending on an allowable base claim.

Claims 51 and 52 both depend on the independent Claim 48. As explained above, the independent Claim 48 is allowable. Accordingly, Claims 51 and 52 are both also allowable as depending on an allowable base claim.

Claims 10 and 30

Within the Office Action, Claims 10 and 30 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Riedel in view of Zadok, and further in view of Noble as applied to Claim 5, and further in view of U.S. Patent No. 5,903,881 to Schrader et al. ("Schrader"). The Applicants respectfully disagree.

Claim 10 depends on the independent Claim 1. As explained above, the independent Claim 1 is allowable. Accordingly, Claim 10 is also allowable as depending on an allowable base claim.

Claim 30 depends on the independent Claim 26. As explained above, the independent Claim 26 is allowable. Accordingly, Claim 30 is also allowable as depending on an allowable base claim.

Claim 13

Within the Office Action, Claim 13 has been rejected under 35 U.S.C. § 103(a) as being unpatentable over Riedel in view of Zadok as applied to Claim 12, and further in view of U.S. Patent No. 5,727,206 to Fish et al. ("Fish"). The Applicants respectfully disagree.

Claim 13 depends on the independent Claim 1. As explained above, the independent Claim 1 is allowable. Accordingly, Claim 13 is also allowable as depending on an allowable base claim.

Claims 16-18, 25, 40, 70, and 71

Within the Office Action, Claims 16-18, 25, 40, 70, and 71 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Riedel in view of Zadok as applied to Claim 15, and further in view of Blaze, "A Cryptographic File System for Unix" ("Blaze"). The Applicants respectfully disagree.

Riedel and Zadok have been characterized above. Blaze is directed to a Cryptographic File System (CFS). Blaze discloses that "Users associate a cryptographic key with the directories they wish to protect. Files in these directories (as well as their pathname components) are transparently encrypted and decrypted with the specified key without further user intervention; cleartext is never stored on a disk or sent to a remote file server." (Blaze, Abstract) Blaze does not disclose an operating system kernel having a virtual node integrated with drivers to encrypt and decrypt data.

In section 2.2, fourth paragraph, cited in the Office Action, Blaze discloses protecting a directory using a set of cryptographic keys, passphrases entered from a keyboard. Blaze discloses, generally, that the passphrases are used to generate several independent keys.

In section 3, third paragraph, cited in the Office Action, Blaze discloses using a passphrase to generate 2 keys. Specifically, Blaze discloses using the first key to compute a bit mask for masking a part of a file block. The result is then encrypted with the second key. The result is not used to generate the second key.

Specifically, Blaze does not disclose taking a pass key *and a data file name* to generate an encrypted data file name key. Blaze does not disclose taking that encrypted file name data key and data file contents *to generate another key*, an encrypted data file contents key. And Blaze does not disclose using that other key and file contents to generate yet another key and also encrypted file contents.

Claims 16-18 and 25 all depend on the independent Claim 1. As explained above, the independent Claim 1 is allowable. Accordingly, Claims 16-18 and 25 are also allowable as depending on an allowable base claim.

Claim 16 is allowable for at least one additional reason. Claim 16 recites, "a key engine to receive a pass key and a data file name to generate an encrypted data file name key, the key engine also to use the encrypted data file name key and data file contents to generate an encrypted data file contents key, the key engine also to encrypt the data file contents with an encrypting data file contents key to generate encrypted data file contents and to encrypt the data file name with an encrypting data file name key to generate an encrypted data file name" (italics added). Not one of Riedel, Zadok, Blaze, and their combination teaches this element. For this additional reason, claim 16 is allowable over Riedel, Zadok, Blaze, and their combination.

Claim 40 depends on the independent Claim 36. As explained above, the independent Claim 36 is allowable. Accordingly, Claim 40 is also allowable as depending on an allowable base claim.

Claim 40 is also allowable for an additional reason. Claim 40 recites, "wherein executing kernel code comprises entering a pass key and a data file name into a first encryption process to produce an encrypted data file name and an encrypted data file name key; and processing the file contents together with the encrypted data file name key to generate an encrypted file contents key and an encrypted file contents." Not one of Riedel, Zadok, and Blaze, either alone or in combination discloses this element. For this additional reason, Claim 40 is allowable.

The independent Claim 70 is directed to a computer system containing an operating system. The computer system of Claim 70 comprises a kernel, a memory, and an encryption key management system. The kernel comprises a virtual node integrated with drivers to encrypt and decrypt data transferred between a memory and a secondary device. The kernel also comprises

an encryption engine to encrypt clear data to generate cipher data. The encryption engine is also to decrypt the cipher data to generate the clear data. The memory is coupled to the encryption engine to store the cipher data and comprises a first logical protected memory to store encrypted file data and a second logical protected memory to store encrypted key data. The encryption key management system is to control access to the encrypted file data and the encrypted key data. The encryption key management system comprises a key engine to receive a pass key and the file name to generate an encrypted file name key, use the encrypted file name key and file contents to generate an encrypted file contents key, and encrypt the file contents with an encrypting file contents key to generate encrypted file contents. Not one of Riedel, Zadok, Blaze, and their combination discloses a kernel that comprises a virtual node integrated with drivers to encrypt and decrypt data transferred between a memory and a secondary device. Not one of Riedel, Zadok, Blaze, and their combination discloses a key engine to receive a pass key and the file name to generate an encrypted file name key, use the encrypted file name key and file contents to generate an encrypted file contents key, and encrypt the file contents with an encrypting file contents key to generate encrypted file contents. For at least these reasons, the independent Claim 70 is allowable over Riedel, Zadok, Blaze, and their combination

The independent Claim 71 is directed to a method of encrypting data. The method of Claim 71 comprises receiving clear data and executing kernel code in an operating system. The kernel code comprises a virtual node integrated with drivers to use a symmetric key to encrypt the clear data to generate cipher data and to use the symmetric key to decrypt the cipher data to generate the clear data. The executing the kernel code comprises entering a pass key and a file name into a first encryption process to produce an encrypted file name and an encrypted file name key and processing the file contents together with the encrypted file name key to generate an encrypted file contents key and encrypted file contents. Not one of Riedel, Zadok, Blaze, and their combination discloses a kernel that comprises a virtual node integrated with drivers to use a symmetric key to encrypt clear data to generate cipher data and to use a symmetric key to decrypt cipher data to generate the clear data. Not one of Riedel, Zadok, Blaze, and their combination discloses that executing kernel code comprises entering a pass key and a file name into a first encryption process to produce an encrypted file name and an encrypted file name key and processing the file contents together with the encrypted file name key to generate an encrypted file contents key and encrypted file contents. For at least these reasons, the independent Claim 71 is allowable over Riedel, Zadok, Blaze, and their combination.

Claims 21, 32, and 44

Within the Office Action, Claims 21, 32, and 44 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Riedel in view of Zadok as applied to Claim 19, and further in view of U.S. Patent No. 6,836,888 to Basu et al. ("Basu"). The Applicants respectfully disagree.

Claim 21 depends on the independent Claim 1. As explained above, the independent Claim 1 is allowable. Accordingly, Claim 21 is also allowable as depending on an allowable base claim.

Claim 32 depends on the independent Claim 26. As explained above, the independent Claim 26 is allowable. Accordingly, Claim 32 is also allowable as depending on an allowable base claim.

Claim 44 depends on the independent Claim 36. As explained above, the independent Claim 36 is allowable. Accordingly, Claim 44 is also allowable as depending on an allowable base claim.

Claims 22-24, 33-35, 45, and 47

Within the Office Action, Claims 22-24, 33-35, 45, and 47 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Riedel in view of Zadok as applied to Claim 19, and further in view of U.S. Patent No. 6,477,545 to LaRue ("LaRue"). The Applicants respectfully disagree.

Claims 22-24 all depend on the independent Claim 1. As explained above, the independent Claim 1 is allowable. Accordingly, Claims 22-24 are all also allowable as depending on an allowable base claim.

Claims 33-35, 45, and 47 all depend on the independent Claim 26. As explained above, the independent Claim 26 is allowable. Accordingly, Claims 33-35, 45, and 47 are all also allowable as depending on an allowable base claim.

Claim 41

Within the Office Action, Claim 41 has been rejected under 35 U.S.C. § 103(a) as being unpatentable over Riedel in view of Zadok as applied to Claim 40, and further in view of Noble. The Applicants respectfully disagree.

Claim 41 depends on the independent Claim 36. As explained above, the independent Claim 36 is allowable. Accordingly, Claim 41 is also allowable as depending on an allowable base claim.

Claims 59 and 60

Within the Office Action, Claims 59 and 60 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Riedel in view of Zadok as applied to Claim 1, and further in view of U.S. Patent No. 6,938,166 to Sarfati et al. ("Sarfati"). The Applicants respectfully disagree.

Claims 59 and 60 both depend on the independent Claim 1. As explained above, the independent Claim 1 is allowable. Accordingly, Claims 59 and 60 are both also allowable as depending on an allowable base claim.

CONCLUSION

For the reasons given above, the Applicants respectfully submit that Claims 1-45, 47-52 and 59-71 are in condition for allowance, and allowance at an early date would be appreciated. If the Examiner has any questions or comments, the Examiner is encouraged to call the undersigned at (408) 530-9700 so that any outstanding issues can be quickly and efficiently resolved.

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: May 18, 2009

By: /Jonathan O. Owens/
Jonathan O. Owens
Reg. No.: 37,902
Attorneys for Applicants